US-PAT-NO: 6577871

DOCUMENT-IDENTIFIER: US 6577871 B1

TITLE: Technique for effectively managing processing loads in a communications arrangement

---------- KWIC ---------

Application Filing Date - AD (1):
19990520

Detailed Description Text - DETX (11):
If the CPU load measurement of any of CEWS(1) through CEWS(N) (or COWS 130) is extremely high, the CEWS (or COWS 130) immediately declares a CPU overload. However, if the **CPU load** measurement exceeds a first predetermined **threshold,** but does not exceed a second higher predetermined **threshold** to warrant immediate defense actions, the CEWS (or COWS 130) needs to wait a "cool-off period," e.g., a predetermined number of measurement periods, during which such a **CPU load** condition persists before declaring a **CPU** overload. Advantageously, with the cool-off period in place, the CEWS (or COWS 130) would not be over-reactive and frequently flip-flop between declaring a **CPU** overload and undeclaring same, especially when the **CPU load** fluctuates around the first predetermined **threshold**.

Detailed Description Text - DETX (17):
The aforementioned CPU overload control routine in accordance with the invention which is run on COWS 130 will now be described. FIG. 3 illustrates such a routine denoted 300 for determining, e.g., the fraction of paging messages that COWS 123 should process to manage its CPU load. Instructed by routine 300, COWS 123 measures the CPU load experienced thereby, .rho..sub.i.sup.COWS, during an i.sup.th measurement period, 1.ltoreq.i, as indicated at step 305. COWS 123 at step 308 compares .rho..sub.i.sup.COWS with a first predetermined COWS **CPU load threshold,** .rho..sub.max.sup.COWS. In this instance, .rho..sub.max.sup.COWS represents the desired maximum CPU load which COWS 123 would sustain, e.g., a load representing 85% of the CPU capacity of COWS 123. If .rho..sub.i.sup.COWS >.rho..sub.max.sup.COWS, a counter, N_COWS, for keeping track of the aforementioned "cool-off" period in terms of measurement periods is triggered, thereby increasing N_COWS by one, as indicated at step 311. Otherwise, if .rho..sub.i.sup.COWS.ltoreq..rho..sub.max.sup.COWS, N_COWS is set to zero, as indicated at step 314.

Detailed Description Text - DETX (18):
Routine 300 proceeds from either step 311 or step 314 to step 317 where .rho..sub.i.sup.COWS is compared with a second, higher predetermined COWS **CPU**

load threshold, .rho..sub.hot.sup.COWS, e.g., a **load** representing 95% of the **CPU** capacity of COWS 123, and where N_COWS is compared with a predetermined number of measurement periods, K.sup.COWS, constituting the required cool-off period. If .rho..sub.i.sup.COWS.gtoreq..rho..sub.hot.sup.COWS or N_COWS.gtoreq.K.sup.COWS, COWS 123 declares a COWS CPU overload, as indicated at step 319. Routine 300 then proceeds from step 319 to step 322 where it is determined whether .rho..sub.i.sup.COWS is smaller than a 100% CPU capacity of COWS 123. If .rho..sub.i.sup.COWS <100%, COWS 123 at step 325 determines the fraction of paging messages to be processed by COWS 123 during the (i+1).sup.th measurement period, .psi..sub.i+1.sup.COWS, as follows: ##EQU1##

Detailed Description Text - DETX (25):
A second CPU overload control routine in accordance with the invention which is run on each of CEWS(1) through CEWS(N), generically represented by CEWS(j), 1.ltoreq.j.ltoreq.N, will now be described. FIG. 4 illustrates such a routine denoted 400 for determining, e.g., the fraction of paging and SMS cell broadcast messages, .psi..sub.i.sup.CEWS(j), and the fraction of MEAS_RES messages, r.sub.i+1.sup.CEWS(j), that CEWS(j) should process during (i+1).sup.th measurement period to manage its CPU load. Instructed by routine 400, CEWS(j) measures the CPU load experienced thereby, .rho..sub.i.sup.CEWS(j), during an i.sup.th measurement period, as indicated at step 405. CEWS(j) at step 408 compares .rho..sub.i.sup.CEWS(j) with a first predetermined CEWS **CPU load threshold,** .rho..sub.max.sup.CEWS. In this instance, .rho..sub.max.sup.CEWS represents the desired maximum CPU load which CEWS(j) would sustain, e.g., a load representing 85% of the CPU capacity of CEWS(j). If .rho..sub.i.sup.CEWS(j) >.rho..sub.max.sup.CEWS, a counter, N_CEWS(j), for keeping track of a second cool-off period in terms of measurement periods is triggered, thereby increasing N_CEWS(j) by one, as indicated at step 411. Otherwise, if .rho..sub.i.sup.CEWS(j).ltoreq..rho..sub.max.sup.CEWS(j), N_CEWS(j) is set to zero, as indicated at step 414.

Detailed Description Text - DETX (26):
Routine 400 proceeds from either step 411 or step 414 to step 417 where .rho..sub.i.sup.CEWS(j) is compared with a second, higher predetermined CEW **CPU load threshold,** .rho..sub.hot.sup.CEWS, e.g., a **load** representing 95% of the **CPU** capacity of CEWS(j), and where N_CEWS(j) is compared with a second predetermined number of measurement periods, K.sup.CEWS, constituting the required second cool-off period. If .rho..sub.i.sup.CEWS(j).gtoreq..rho..sub.hot.sup.CEWS or N_CEWS(j).gtoreq.K.sup.CEWS, CEWS(j) declares a CEWS(j) CPU overload, as indicated at step 419. Routine 400 then proceeds from step 419 to step 422 where it is determined whether .rho..sub.i.sup.CEWS(j) is smaller than a 100% CPU capacity of CEWS(j). If .rho..sub.i.sup.CEWS(j) <100%, CEWS(j) at step 425 determines .psi..sub.i+1.sup.CEWS(j) as follows: ##EQU4##